University of Saskatchewan
Department of Computer Science

44/50

CMPT 374 Theory and Application of Databases
Midterm Examination

Wednesday March 5, 2003 11:30 a.m. - 12:20 p.m. (50 minutes)

Closed Book

This exam has 7 pages. It consists of two parts. Part 1 has 18 multiple-choice questions. Part 2 has 3 questions.

16
8
14
6
44

Instructions

Please answer all the multiple-choice questions in Part 1 on the opscan sheet provided. Answer all the questions in Part 2 on this booklet itself. If you need additional writing material or have any questions, please raise your hand. Do not leave your seat until you are ready to hand in your exam.

16/18

Part 1 (Multiple Choice) (18 marks)

1. Which one of the following is an **advantage** of using a database management system?
   A. Control of data redundancy
   B. Data consistency
   C. More information from the same amount of data
   D. Enforcement of standards
   E. All of the above

2. What one of the following statements is **false** about a database?
   A. The external level is the user's view of the database
   B. The conceptual level is the community view of the database
   C. Physical data independence refers to the immunity of the external schema to changes in the internal schema
   D. The external/conceptual mapping relates the external schema to the conceptual schema
   E. None of the above

3. Which of the following operations is **not** associated with a data manipulation language?
   A. Insertion of new data into the database
   B. Modification of data stored in the database
   C. Retrieval of data contained in the database
   D. Creation of tables and views
   E. None of the above

4. The database server does **not** typically perform which of the following functions in a client-server database system?
   A. Processing of queries/updates and transmitting results to clients
   B. Authorization of users
   C. Management of the user interface
   D. Maintenance of the system catalogs
   E. None of the above

5. What is a relational schema?
   A. A named relation defined by a set of attribute and domain name pairs
   B. A set of relations each with a distinct name
   C. An attribute, or set of attributes, that uniquely identifies a tuple within a relation
   D. A collection of normalized relations with distinct names
   E. None of the above

6. Which one of the following statements is **true**?
   A. The primary key of one relation can also be a foreign key of another relation
   B. Candidate keys are both unique and irreducible
   C. No component of a primary key can be null
   D. A primary key may contain more than one attribute
   E. All of the above

7. Which one of the following states that a **foreign key** must either match a primary key value in another relation or it must be null?
   A. Entity integrity rule
   B. Referential integrity rule
   C. Composite attribute rule
   D. Enterprise constraint
   E. None of the above

8. A **tuple** is best described as:
   A. A row of data in a relation
   B. A relation within a database
   C. An attribute within a row
   D. A set of rows selected from a table or group of tables
   E. None of the above

9. The **degree** of a relation is defined as:
   A. The number of tuples it contains
   B. The number of records it contains
   C. The number of attributes it contains
   D. The number of different data types it contains
   E. None of the above

10. With regard to a relation, which one of the following statements is **false**?
    A. The order of attributes within a relation is unimportant
    B. Each cell of the relation contains one or more values
    C. Each attribute has a distinct name
    D. Each tuple within the relation is distinct (i.e., there are no duplicate tuples)
    E. None of the above

11. The statement "R and S are **union-compatible**" means:
    A. R and S have the same number of tuples
    B. R and S have the same number of attributes
    C. Either R is a subset of S or S is a subset of R
    D. R and S have the same relational schema
    E. None of the above

12. The **difference** between two union-compatible relations A and B, denoted A-B, contains:
    A. The set of tuples in both A and B
    B. The set of tuples in B but not in A
    C. The set of tuples in A but not in B
    D. The same set of tuples as B-A.
    E. None of the above

13. Which one of the following statements is true about the **union** of two union-compatible relations A and B?
    A. It contains tuples that are in A, and which are also in B
    B. It contains tuples that are in A, or in B, but not necessarily the both of them
    C. If A and B contains an identical tuple, the union contains only one copy of the tuple; i.e., the duplicate is removed.
    D. The degree of resulting relation is the same as the degree of A
    E. All of the above

14. Which one of the following statements is true about the **cartesian** product of two relations A (cardinality *ca*, degree *da*), and B (cardinality *cb*, degree *db*):
    A. A and B must be union-compatible
    B. The cardinality of the cartesian product is ($ca * cb$) and the degree is ($da + db$)
    C. The cardinality of the cartesian product is ($ca + cb$) and the degree is ($da * db$)
    D. The cardinality of the cartesian product is less than ($ca + cb$), but the degree is ($da + db$)
    E. None of the above

15. Which one of the following statements is true about the **semi-join** between two relations A and B, $A \triangleright B$:
    A. There are no attributes of B in the semi-join
    B. All the attributes of B are present in the semi-join
    C. Only the common attributes of A and B are present in the semi-join
    D. All the attributes of A are present in the semi-join
    E. None of the above

16. Which one of the following statements is **true** about **SQL**?
    A. It has a Data Definition Language component
    B. It has a Data Manipulation Language component
    C. It is an international standard
    D. It is a non-procedural language
    E. All of the above

17. Which one of the following is **not** true about the SQL aggregate functions?
    A. SUM and AVG may be used on numeric fields only
    B. COUNT, MIN, and MAX may be used on both numeric and non-numeric fields
    C. COUNT counts all the rows of a column after eliminating rows with null columns
    D. COUNT(*) counts all the rows of a table, including duplicates
    E. None of the above

18. Which one of the following statements is **true** about the condition: `address LIKE 'H%'`
   A. It refers to all addresses having two characters and beginning with an H
   B. It refers to all addresses having an H anywhere in the string
   C. It refers to all addresses beginning with an H, but followed by any amount of characters which may or may not contain an H
   D. It refers to all addresses that do not begin with an H
   E. None of the above

Part 2

1. Consider the following set of relations that keeps information about possible journeys from Saskatoon to a variety of destinations (primary keys are highlighted in bold and underlined):

Operator (**opCode**, opName)
Destination (**destinationCode**, destinationName, distance)
Journey (**opCode**, **destinationCode**, price)

where Operator — contains the name of the operator who services the journey.
Destination — contains destination details and *distance* is the distance in kilometers from Saskatoon.
Journey — records the price of an adult fare from Saskatoon to the given destination by a specified operator. Note that several operators may operate over the same route (e.g., airline, bus company, etc.).

Formulate the following queries using the relational algebra:

(a) List the names of all destinations.

$\Pi_{destinationName}$ (Destination) ✓ 1/1

(b) Find the names of all destinations within 20 kilometers.

$\Pi_{destinationName}$ ($\sigma_{distance <= 20}$ (Destination)) 2/2

8/10

(c) List the names of all operators with at least one journey priced at under $20.00.

Π_{opName} ($\sigma_{price < 20.00 \text{ AND } Operator.opCode = Journey.opCode}$ ((Operator) × (Journey))) 2/2

(d) List the names of all operators and prices of journeys to 'Prince Albert'.

$\Pi_{opName, price}$ ($\sigma_{destinationName = 'Prince Albert'}$ ((Operator) NJ (Journey) NJ (Destination))

Using NJ as the natural join ✓ 3/3

(e) List the names of all destinations that do not have any operators.

$\Pi_{destinationName}(\sigma_{operator = \ldots}$

9/2

[10 marks]

2. The relational schema shown below is part of a hospital database. The primary keys are highlighted in bold and underlined, except for the *Prescribed* relation. The *Contains* relation only contains patients who are currently in the hospital.

Patient (**patientNo**, patName, patAddress, DOB)
Ward (**wardNo**, wardName, wardType, noOfBeds)
Contains (**patientNo, wardNo, admissionDate**)
Drug (**drugNo**, drugName, costPerUnit)
Prescribed (patientNo, drugNo, unitsPerDay, startDate, finishDate)

(a) Choose a primary key for the *Prescribed* relation.

patientNo, drugNo ✓

1/1

(b) Write the SQL statement to create the *Prescribed* relation, given that at most 10 units of any drug can be administered in any given day. Choose any reasonable data type for the attributes. The update policy for the foreign keys is CASCADE and the delete policy is NO ACTION.

```
CREATE TABLE Prescribed (
    patientNo       int [6]     NOT NULL,
    drugNo          int [6]     NOT NULL,
    unitsPerDay     int [2]     DEFAULT 0,
    startDate       date        NOT NULL,
    endDate         date        NOT NULL

PRIMARY KEY ( patientNo, drugNo )
CONSTRAINT tooManyUnits,
    CHECK ( unitsPerDay <= 10)

FOREIGN KEY (patientNo) REFERENCES Patient ON UPDATE CASCADE,
                                   ON DELETE NO ACTION

FOREIGN KEY (drugNo) REFERENCES Drug ON UPDATE CASCADE,
                                 ON DELETE NO ACTION ; 5
)
```

4/4

(c) Write SQL statements for the following queries:

(i) List the details of all patients, in ascending order by patName.

```
SELECT  *
FROM    Patient
ORDER BY  patName ;
```

2/2

(ii) List the patientNo and patName of all the patients admitted today.

```
SELECT p.patientNo, p.patName
FROM   patient p , contains c
WHERE  p.patientNo = c.patientNo AND c.admissionDate = 'Mar-5-02' ;
```

3/3

(iii) What are the maximum, minimum, and average number of beds in a ward? Create appropriate column headings for the results displayed.

```
SELECT   MAX(no of beds) AS MaxNumberofBeds , MIN(noOfBeds) AS minNumberofBeds,
         AVG(no of beds) As avgNumberofBeds
FROM     Ward ;
```

2/2

14/15

(iv) For each ward that admitted more than 10 patients today, list the ward number, ward type, and number of beds in each ward.

```
SELECT  w.wardNo, w.wardType , w.no of Beds
FROM    Ward
WHERE   WardNo IN ( SELECT  wardNo
                    FROM    Contains
                    WHERE   COUNT(patientNo) > 10 ) ;
```

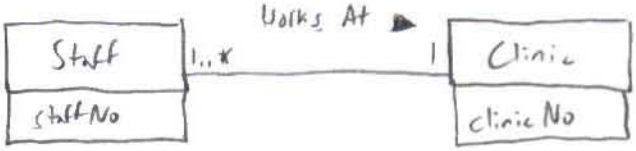
today?

2/3 ✗

[15 marks]

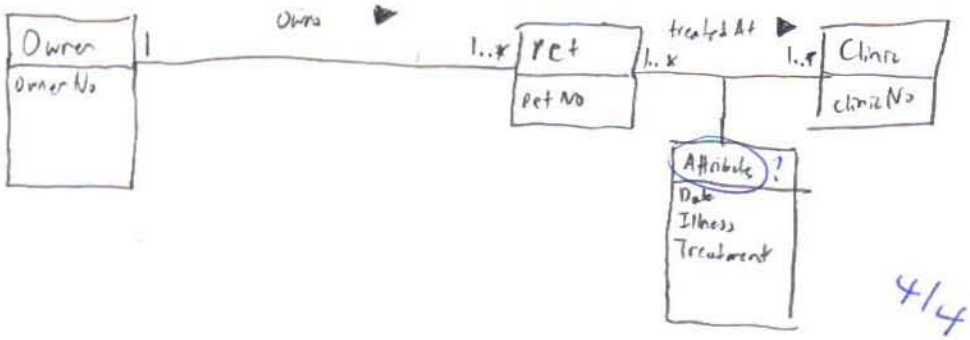
3. Represent each of the following requirements with an Entity-Relationship diagram:

(a) A company called Perfect Pets runs a number of clinics. A clinic has many staff and a member of staff manages at most one clinic (not all staff manage clinics). Each clinic has a unique clinic number (clinicNo) and each member of staff has a unique staff number (staffNo).



2/3

(b) A pet owner can own one or more pets, and a pet can be treated at any clinic. Each time a pet is treated, the date, illness, and treatment are recorded. Each owner has a unique owner number (ownerNo) and each pet has a unique pet number (petNo).



6/7

4/4

[7 marks]

[Total Marks 50]

End of Examination